Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

**Программное обеспечение встроенных систем**

**Отчёт**

По лабораторной работе №4

Тема работы:

Использование Bluetooth LE модуля HM-10 с STM-32

Выполнили: Шуляк А.В.

Дранкевич А.А.

Проверил: Леванцевич В. А.

Минск 2023

1. Чип HM-10

Модуль HM-10 представляет собой Bluetooth-модуль, который широко используется для беспроводной связи в различных электронных устройствах. Основные характеристики и особенности чипа HM-10:

* Работа в режимах Master и Slave
* Поддержка U(S)ART
* Питание - 5 (3.3) В
* Потребление модуля - 8.5 мА в активном состоянии
* Поддержка Bluetooth Low Energy
* Поддержка AT-команд для управления

2. Настройка оборудования

Перед подключением модуля к STM32 его необходимо сбросить до заводских настроек для обеспечения гарантии его корректной работы. Для этого необходимо подключить модуль через USB-переходник, подключиться к соответствующему последовательному виртуальному порту и отправить команду «AT+RENEW\r\n», ожидаемый ответ модуля - «OK+RENEW\r\n».

Для непосредственного подключения модуля к STM32 необходимо подключить пины модуля к плате следующим образом:

* VDD к пину +3.3V (+5V) STM32
* GND к соответствующему пину STM32
* RX к пину PA2
* TX к пину PA10

После чего необходимо настроить USART и DMA для дуплексной передачи нданных между микроконтроллером и модулем HM-10, как это указано в Приложении А, файл с именем Lab4.ioc

3. Работа модуля

Для работы модуля HM-10 его необходимо корректно настроить, для этого с STM32 нужно отправить по интерфейсу USART на модуль следующие команды:

1. TRANSMIT(huart1, "AT+MODE0\r\n"); – проверка работоспособности модуля, ожидаемый ответ - "OK\r\n"
2. TRANSMIT(huart1, "AT+MODE0\r\n"); – установка режима работы модуля – в 0-ом режиме модуль реагирует на управляющие команды только без наличия подключённого к нему по Bluetooth узла, при подключении переходит в режим пересылки данных
3. TRANSMIT(huart1, "AT+NAMESTUPID\_STM\r\n"); – установка имени для Bluetooth-маячка, под которым он будет обнаружен клиентами
4. TRANSMIT(huart1, "AT+RESET\r\n"); – перезагрузка модуля, необходима для того, чтобы все внесённые изменения в настройках вступили в силу

После выполнения этих шагов модуль начинает корректно реагировать на попытки подключения (устройство с Android и приложением BLE Terminal). В рамках лабораторной работы был реализован «эхо-сервер», который ожидает от клиента отправки 4-символьной строковой последовательности, а затем отправляет её клиенту назад. Пересылка строк произвольной длины затруднена, поскольку данный модуль не предназначен для отправки большого количества данных между подключёнными узлами, а служит именно для обмена короткими сообщениями.

**ПРИЛОЖЕНИЕ А**

**Исходный код программы**

**Main.c**

**#include "main.h"**

**#include <stdio.h>**

**#include <string.h>**

**#define true (bool)1;**

**#define false (bool)0;**

**#define UART\_TIMEOUT 1000**

**#define CHECK\_ON(what) (strncmp((char \*)buffer, (what), sizeof (what)) == 0)**

**#define TRANSMIT(uart, msg) err = HAL\_UART\_Transmit(&(uart), (unsigned char \*)(msg), sizeof (msg) - 1, UART\_TIMEOUT);\**

**if (err != HAL\_OK) {\**

**while (1) {\**

**HAL\_GPIO\_WritePin(LD2\_GPIO\_Port, LD2\_Pin, GPIO\_PIN\_RESET); HAL\_Delay(100);\**

**HAL\_GPIO\_WritePin(LD2\_GPIO\_Port, LD2\_Pin, GPIO\_PIN\_SET);HAL\_Delay(100);\**

**}\**

**}**

**int myreceive(UART\_HandleTypeDef \*uart, unsigned char \*buffer, int required) {**

**int i = 0;**

**while ((HAL\_UART\_Receive(uart, buffer + i, 1, UART\_TIMEOUT) == HAL\_OK) && (i < required)) {i++; };**

**return i;**

**}**

**#define RECIEVE(uart, msg) if ((myreceive(&(uart), buffer, 256) == 0) || !CHECK\_ON((msg))) {\**

**while (1) {\**

**HAL\_GPIO\_WritePin(LD2\_GPIO\_Port, LD2\_Pin, GPIO\_PIN\_RESET); HAL\_Delay(100);\**

**HAL\_GPIO\_WritePin(LD2\_GPIO\_Port, LD2\_Pin, GPIO\_PIN\_SET);HAL\_Delay(100);\**

**}\**

**}**

**#define RECIEVEN(uart, num) if (myreceive(&(uart), buffer, num) != num) {\**

**while (1) {\**

**HAL\_GPIO\_WritePin(LD2\_GPIO\_Port, LD2\_Pin, GPIO\_PIN\_RESET); HAL\_Delay(100);\**

**HAL\_GPIO\_WritePin(LD2\_GPIO\_Port, LD2\_Pin, GPIO\_PIN\_SET);HAL\_Delay(100);\**

**}\**

**}**

**UART\_HandleTypeDef huart1;**

**UART\_HandleTypeDef huart2;**

**DMA\_HandleTypeDef hdma\_usart1\_rx;**

**void SystemClock\_Config(void);**

**static void MX\_GPIO\_Init(void);**

**static void MX\_DMA\_Init(void);**

**static void MX\_USART2\_UART\_Init(void);**

**static void MX\_USART1\_UART\_Init(void);**

**volatile int listen\_mode = 0;**

**volatile int bpos = 0;**

**unsigned char buffer[256];**

**unsigned char buffer\_a[256];**

**unsigned char buffer\_b[256];**

**unsigned char \*pb\_a = buffer\_a;**

**unsigned char \*pb\_b = buffer\_b;**

**void HAL\_UART\_RxHalfCpltCallback(UART\_HandleTypeDef \*huart) {}**

**void HAL\_UART\_RxCpltCallback(UART\_HandleTypeDef \*huart)**

**{**

**HAL\_GPIO\_TogglePin (LD2\_GPIO\_Port, LD2\_Pin); // toggle PA0**

**bpos=2;**

**//HAL\_UART\_Transmit(&(huart1), (unsigned char \*)(buffer),6, UART\_TIMEOUT);**

**HAL\_UART\_Transmit(&(huart2), (unsigned char \*)(buffer),6, UART\_TIMEOUT);**

**if (listen\_mode) {**

**HAL\_UART\_Receive\_DMA(&huart1, buffer, 6);**

**}**

**}**

**int main(void)**

**{**

**HAL\_Init();**

**SystemClock\_Config();**

**MX\_GPIO\_Init();**

**MX\_DMA\_Init();**

**MX\_USART2\_UART\_Init();**

**MX\_USART1\_UART\_Init();**

**HAL\_GPIO\_WritePin(LD2\_GPIO\_Port, LD2\_Pin, GPIO\_PIN\_SET);**

**HAL\_StatusTypeDef err;**

**TRANSMIT(huart2, "AT\r\n");**

**TRANSMIT(huart1, "AT\r\n");**

**bpos = 0; // Clear receive flag**

**err = HAL\_UART\_Receive\_DMA (&huart1, buffer, sizeof ("OK\r\n") - 1); // request data**

**while (bpos != 2) {};**

**TRANSMIT(huart2, "AT+MODE0\r\n");**

**TRANSMIT(huart1, "AT+MODE0\r\n");**

**bpos = 0; // Clear receive flag**

**err = HAL\_UART\_Receive\_DMA (&huart1, buffer, sizeof ("OK+SET:0\r\n") - 1); // request data**

**while (bpos != 2) {};**

**TRANSMIT(huart2, "AT+NAMESTUPID\_STM\r\n");**

**TRANSMIT(huart1, "AT+NAMESTUPID\_STM\r\n");**

**bpos = 0;**

**err = HAL\_UART\_Receive\_DMA (&huart1, buffer, sizeof ("OK+SET:STUPID\_STM\r\n") - 1);**

**while (bpos != 2) {};**

**TRANSMIT(huart2, "AT+NAMEI\_HATE\_STM\r\n");**

**TRANSMIT(huart1, "AT+NAMEI\_HATE\_STM\r\n");**

**bpos = 0;**

**err = HAL\_UART\_Receive\_DMA (&huart1, buffer, sizeof ("OK+SET:I\_HATE\_STM\r\n") - 1);**

**while (bpos != 2) {};**

**TRANSMIT(huart2, "AT+RESET\r\n");**

**TRANSMIT(huart1, "AT+RESET\r\n");**

**bpos = 0;**

**err = HAL\_UART\_Receive\_DMA (&huart1, buffer, sizeof ("OK+RESET\r\n") - 1);**

**while (bpos != 2) {};**

**listen\_mode = 1;**

**HAL\_UART\_Receive\_DMA(&huart1, buffer, 6);**

**while (1) { };**

**Error\_Handler();**

**TRANSMIT(huart2, "AT\r\n");**

**TRANSMIT(huart1, "AT\r\n");**

**HAL\_Delay(500);**

**RECIEVE(huart1, "OK");**

**TRANSMIT(huart2, "AT+MODE0\r\n");**

**TRANSMIT(huart1, "AT+MODE0\r\n");**

**HAL\_Delay(500);**

**RECIEVE(huart1, "OK+SET:0");**

**TRANSMIT(huart2, "AT+NAMESTUPID\_STM\r\n");**

**TRANSMIT(huart1, "AT+NAMESTUPID\_STM\r\n");**

**HAL\_Delay(500);**

**RECIEVE(huart1, "OK+SET:STUPID\_STM");**

**TRANSMIT(huart2, "AT+RESET\r\n");**

**TRANSMIT(huart1, "AT+RESET\r\n");**

**HAL\_Delay(500);**

**RECIEVE(huart1, "OK+RESET");**

**while (1) {}**

**}**

**void SystemClock\_Config(void)**

**{**

**RCC\_OscInitTypeDef RCC\_OscInitStruct = {0};**

**RCC\_ClkInitTypeDef RCC\_ClkInitStruct = {0};**

**RCC\_OscInitStruct.OscillatorType = RCC\_OSCILLATORTYPE\_HSI;**

**RCC\_OscInitStruct.HSIState = RCC\_HSI\_ON;**

**RCC\_OscInitStruct.HSICalibrationValue = RCC\_HSICALIBRATION\_DEFAULT;**

**RCC\_OscInitStruct.PLL.PLLState = RCC\_PLL\_ON;**

**RCC\_OscInitStruct.PLL.PLLSource = RCC\_PLLSOURCE\_HSI\_DIV2;**

**RCC\_OscInitStruct.PLL.PLLMUL = RCC\_PLL\_MUL16;**

**if (HAL\_RCC\_OscConfig(&RCC\_OscInitStruct) != HAL\_OK)**

**{**

**Error\_Handler();**

**}**

**RCC\_ClkInitStruct.ClockType = RCC\_CLOCKTYPE\_HCLK|RCC\_CLOCKTYPE\_SYSCLK**

**|RCC\_CLOCKTYPE\_PCLK1|RCC\_CLOCKTYPE\_PCLK2;**

**RCC\_ClkInitStruct.SYSCLKSource = RCC\_SYSCLKSOURCE\_PLLCLK;**

**RCC\_ClkInitStruct.AHBCLKDivider = RCC\_SYSCLK\_DIV1;**

**RCC\_ClkInitStruct.APB1CLKDivider = RCC\_HCLK\_DIV2;**

**RCC\_ClkInitStruct.APB2CLKDivider = RCC\_HCLK\_DIV1;**

**if (HAL\_RCC\_ClockConfig(&RCC\_ClkInitStruct, FLASH\_LATENCY\_2) != HAL\_OK)**

**{**

**Error\_Handler();**

**}**

**}**

**static void MX\_USART1\_UART\_Init(void)**

**{**

**huart1.Instance = USART1;**

**huart1.Init.BaudRate = 9600;**

**huart1.Init.WordLength = UART\_WORDLENGTH\_8B;**

**huart1.Init.StopBits = UART\_STOPBITS\_1;**

**huart1.Init.Parity = UART\_PARITY\_NONE;**

**huart1.Init.Mode = UART\_MODE\_TX\_RX;**

**huart1.Init.HwFlowCtl = UART\_HWCONTROL\_NONE;**

**huart1.Init.OverSampling = UART\_OVERSAMPLING\_16;**

**if (HAL\_UART\_Init(&huart1) != HAL\_OK)**

**{**

**Error\_Handler();**

**}**

**}**

**static void MX\_USART2\_UART\_Init(void)**

**{**

**huart2.Instance = USART2;**

**huart2.Init.BaudRate = 9600;**

**huart2.Init.WordLength = UART\_WORDLENGTH\_8B;**

**huart2.Init.StopBits = UART\_STOPBITS\_1;**

**huart2.Init.Parity = UART\_PARITY\_NONE;**

**huart2.Init.Mode = UART\_MODE\_TX\_RX;**

**huart2.Init.HwFlowCtl = UART\_HWCONTROL\_NONE;**

**huart2.Init.OverSampling = UART\_OVERSAMPLING\_16;**

**if (HAL\_UART\_Init(&huart2) != HAL\_OK)**

**{**

**Error\_Handler();**

**}**

**}**

**static void MX\_DMA\_Init(void)**

**{**

**\_\_HAL\_RCC\_DMA1\_CLK\_ENABLE();**

**HAL\_NVIC\_SetPriority(DMA1\_Channel5\_IRQn, 0, 0);**

**HAL\_NVIC\_EnableIRQ(DMA1\_Channel5\_IRQn);**

**}**

**static void MX\_GPIO\_Init(void)**

**{**

**GPIO\_InitTypeDef GPIO\_InitStruct = {0};**

**\_\_HAL\_RCC\_GPIOC\_CLK\_ENABLE();**

**\_\_HAL\_RCC\_GPIOD\_CLK\_ENABLE();**

**\_\_HAL\_RCC\_GPIOA\_CLK\_ENABLE();**

**\_\_HAL\_RCC\_GPIOB\_CLK\_ENABLE();**

**HAL\_GPIO\_WritePin(GPIOA, LED\_Pin|LD2\_Pin, GPIO\_PIN\_RESET);**

**GPIO\_InitStruct.Pin = B1\_Pin;**

**GPIO\_InitStruct.Mode = GPIO\_MODE\_IT\_RISING;**

**GPIO\_InitStruct.Pull = GPIO\_NOPULL;**

**HAL\_GPIO\_Init(B1\_GPIO\_Port, &GPIO\_InitStruct);**

**GPIO\_InitStruct.Pin = BT\_Status\_Pin;**

**GPIO\_InitStruct.Mode = GPIO\_MODE\_INPUT;**

**GPIO\_InitStruct.Pull = GPIO\_NOPULL;**

**HAL\_GPIO\_Init(BT\_Status\_GPIO\_Port, &GPIO\_InitStruct);**

**GPIO\_InitStruct.Pin = LED\_Pin|LD2\_Pin;**

**GPIO\_InitStruct.Mode = GPIO\_MODE\_OUTPUT\_PP;**

**GPIO\_InitStruct.Pull = GPIO\_NOPULL;**

**GPIO\_InitStruct.Speed = GPIO\_SPEED\_FREQ\_LOW;**

**HAL\_GPIO\_Init(GPIOA, &GPIO\_InitStruct);**

**HAL\_NVIC\_SetPriority(EXTI15\_10\_IRQn, 0, 0);**

**HAL\_NVIC\_EnableIRQ(EXTI15\_10\_IRQn);**

**}**

**void Error\_Handler(void)**

**{**

**\_\_disable\_irq();**

**while (1) { }**

**}**

**Lab4.ioc**

**#MicroXplorer Configuration settings - do not modify**

**CAD.formats=**

**CAD.pinconfig=**

**CAD.provider=**

**Dma.Request0=USART1\_RX**

**Dma.RequestsNb=1**

**Dma.USART1\_RX.0.Direction=DMA\_PERIPH\_TO\_MEMORY**

**Dma.USART1\_RX.0.Instance=DMA1\_Channel5**

**Dma.USART1\_RX.0.MemDataAlignment=DMA\_MDATAALIGN\_BYTE**

**Dma.USART1\_RX.0.MemInc=DMA\_MINC\_ENABLE**

**Dma.USART1\_RX.0.Mode=DMA\_CIRCULAR**

**Dma.USART1\_RX.0.PeriphDataAlignment=DMA\_PDATAALIGN\_BYTE**

**Dma.USART1\_RX.0.PeriphInc=DMA\_PINC\_DISABLE**

**Dma.USART1\_RX.0.Priority=DMA\_PRIORITY\_VERY\_HIGH**

**Dma.USART1\_RX.0.RequestParameters=Instance,Direction,PeriphInc,MemInc,PeriphDataAlignment,MemDataAlignment,Mode,Priority**

**File.Version=6**

**GPIO.groupedBy=Group By Peripherals**

**KeepUserPlacement=false**

**Mcu.CPN=STM32F103RBT6**

**Mcu.Family=STM32F1**

**Mcu.IP0=DMA**

**Mcu.IP1=NVIC**

**Mcu.IP2=RCC**

**Mcu.IP3=SYS**

**Mcu.IP4=USART1**

**Mcu.IP5=USART2**

**Mcu.IPNb=6**

**Mcu.Name=STM32F103R(8-B)Tx**

**Mcu.Package=LQFP64**

**Mcu.Pin0=PC13-TAMPER-RTC**

**Mcu.Pin1=PC14-OSC32\_IN**

**Mcu.Pin10=PA9**

**Mcu.Pin11=PA10**

**Mcu.Pin12=PA13**

**Mcu.Pin13=PA14**

**Mcu.Pin14=PB3**

**Mcu.Pin15=VP\_SYS\_VS\_Systick**

**Mcu.Pin2=PC15-OSC32\_OUT**

**Mcu.Pin3=PD0-OSC\_IN**

**Mcu.Pin4=PD1-OSC\_OUT**

**Mcu.Pin5=PA1**

**Mcu.Pin6=PA2**

**Mcu.Pin7=PA3**

**Mcu.Pin8=PA4**

**Mcu.Pin9=PA5**

**Mcu.PinsNb=16**

**Mcu.ThirdPartyNb=0**

**Mcu.UserConstants=**

**Mcu.UserName=STM32F103RBTx**

**MxCube.Version=6.9.1**

**MxDb.Version=DB.6.0.91**

**NVIC.BusFault\_IRQn=true\:0\:0\:false\:false\:true\:true\:false\:false**

**NVIC.DMA1\_Channel5\_IRQn=true\:0\:0\:false\:false\:true\:false\:true\:true**

**NVIC.DebugMonitor\_IRQn=true\:0\:0\:false\:false\:true\:true\:false\:false**

**NVIC.EXTI15\_10\_IRQn=true\:0\:0\:false\:false\:true\:true\:true\:true**

**NVIC.ForceEnableDMAVector=true**

**NVIC.HardFault\_IRQn=true\:0\:0\:false\:false\:true\:true\:false\:false**

**NVIC.MemoryManagement\_IRQn=true\:0\:0\:false\:false\:true\:true\:false\:false**

**NVIC.NonMaskableInt\_IRQn=true\:0\:0\:false\:false\:true\:true\:false\:false**

**NVIC.PendSV\_IRQn=true\:0\:0\:false\:false\:true\:true\:false\:false**

**NVIC.PriorityGroup=NVIC\_PRIORITYGROUP\_4**

**NVIC.SVCall\_IRQn=true\:0\:0\:false\:false\:true\:true\:false\:false**

**NVIC.SysTick\_IRQn=true\:0\:0\:false\:false\:true\:true\:true\:false**

**NVIC.USART1\_IRQn=true\:0\:0\:false\:false\:true\:true\:true\:true**

**NVIC.UsageFault\_IRQn=true\:0\:0\:false\:false\:true\:true\:false\:false**

**PA1.GPIOParameters=GPIO\_Label**

**PA1.GPIO\_Label=BT\_Status**

**PA1.Locked=true**

**PA1.Signal=GPIO\_Input**

**PA10.GPIOParameters=GPIO\_Mode**

**PA10.GPIO\_Mode=GPIO\_MODE\_INPUT**

**PA10.Mode=Asynchronous**

**PA10.Signal=USART1\_RX**

**PA13.GPIOParameters=GPIO\_Label**

**PA13.GPIO\_Label=TMS**

**PA13.Locked=true**

**PA13.Mode=Serial\_Wire**

**PA13.Signal=SYS\_JTMS-SWDIO**

**PA14.GPIOParameters=GPIO\_Label**

**PA14.GPIO\_Label=TCK**

**PA14.Locked=true**

**PA14.Mode=Serial\_Wire**

**PA14.Signal=SYS\_JTCK-SWCLK**

**PA2.GPIOParameters=GPIO\_Speed,GPIO\_PuPd,GPIO\_Label,GPIO\_Mode**

**PA2.GPIO\_Label=USART\_TX**

**PA2.GPIO\_Mode=GPIO\_MODE\_AF\_PP**

**PA2.GPIO\_PuPd=GPIO\_NOPULL**

**PA2.GPIO\_Speed=GPIO\_SPEED\_FREQ\_LOW**

**PA2.Locked=true**

**PA2.Mode=Asynchronous**

**PA2.Signal=USART2\_TX**

**PA3.GPIOParameters=GPIO\_Speed,GPIO\_PuPd,GPIO\_Label,GPIO\_Mode**

**PA3.GPIO\_Label=USART\_RX**

**PA3.GPIO\_Mode=GPIO\_MODE\_AF\_PP**

**PA3.GPIO\_PuPd=GPIO\_NOPULL**

**PA3.GPIO\_Speed=GPIO\_SPEED\_FREQ\_LOW**

**PA3.Locked=true**

**PA3.Mode=Asynchronous**

**PA3.Signal=USART2\_RX**

**PA4.GPIOParameters=GPIO\_Label**

**PA4.GPIO\_Label=LED**

**PA4.Locked=true**

**PA4.Signal=GPIO\_Output**

**PA5.GPIOParameters=GPIO\_Speed,GPIO\_PuPd,GPIO\_Label,GPIO\_ModeDefaultOutputPP**

**PA5.GPIO\_Label=LD2 [Green Led]**

**PA5.GPIO\_ModeDefaultOutputPP=GPIO\_MODE\_OUTPUT\_PP**

**PA5.GPIO\_PuPd=GPIO\_NOPULL**

**PA5.GPIO\_Speed=GPIO\_SPEED\_FREQ\_LOW**

**PA5.Locked=true**

**PA5.Signal=GPIO\_Output**

**PA9.Mode=Asynchronous**

**PA9.Signal=USART1\_TX**

**PB3.GPIOParameters=GPIO\_Label**

**PB3.GPIO\_Label=SWO**

**PB3.Locked=true**

**PB3.Signal=SYS\_JTDO-TRACESWO**

**PC13-TAMPER-RTC.GPIOParameters=GPIO\_PuPd,GPIO\_Label**

**PC13-TAMPER-RTC.GPIO\_Label=B1 [Blue PushButton]**

**PC13-TAMPER-RTC.GPIO\_PuPd=GPIO\_NOPULL**

**PC13-TAMPER-RTC.Locked=true**

**PC13-TAMPER-RTC.Signal=GPXTI13**

**PC14-OSC32\_IN.Locked=true**

**PC14-OSC32\_IN.Mode=LSE-External-Oscillator**

**PC14-OSC32\_IN.Signal=RCC\_OSC32\_IN**

**PC15-OSC32\_OUT.Locked=true**

**PC15-OSC32\_OUT.Mode=LSE-External-Oscillator**

**PC15-OSC32\_OUT.Signal=RCC\_OSC32\_OUT**

**PD0-OSC\_IN.Locked=true**

**PD0-OSC\_IN.Mode=HSE-External-Clock-Source**

**PD0-OSC\_IN.Signal=RCC\_OSC\_IN**

**PD1-OSC\_OUT.Locked=true**

**PD1-OSC\_OUT.Mode=HSE-External-Clock-Source**

**PD1-OSC\_OUT.Signal=RCC\_OSC\_OUT**

**PinOutPanel.RotationAngle=0**

**ProjectManager.AskForMigrate=true**

**ProjectManager.BackupPrevious=false**

**ProjectManager.CompilerOptimize=6**

**ProjectManager.ComputerToolchain=false**

**ProjectManager.CoupleFile=false**

**ProjectManager.CustomerFirmwarePackage=**

**ProjectManager.DefaultFWLocation=true**

**ProjectManager.DeletePrevious=true**

**ProjectManager.DeviceId=STM32F103RBTx**

**ProjectManager.FirmwarePackage=STM32Cube FW\_F1 V1.8.5**

**ProjectManager.FreePins=false**

**ProjectManager.HalAssertFull=false**

**ProjectManager.HeapSize=0x200**

**ProjectManager.KeepUserCode=true**

**ProjectManager.LastFirmware=true**

**ProjectManager.LibraryCopy=0**

**ProjectManager.MainLocation=Core/Src**

**ProjectManager.NoMain=false**

**ProjectManager.PreviousToolchain=**

**ProjectManager.ProjectBuild=false**

**ProjectManager.ProjectFileName=Lab4.ioc**

**ProjectManager.ProjectName=Lab4**

**ProjectManager.ProjectStructure=**

**ProjectManager.RegisterCallBack=**

**ProjectManager.StackSize=0x400**

**ProjectManager.TargetToolchain=MDK-ARM V5.32**

**ProjectManager.ToolChainLocation=**

**ProjectManager.UAScriptAfterPath=**

**ProjectManager.UAScriptBeforePath=**

**ProjectManager.UnderRoot=false**

**ProjectManager.functionlistsort=1-SystemClock\_Config-RCC-false-HAL-false,2-MX\_GPIO\_Init-GPIO-false-HAL-true,3-MX\_DMA\_Init-DMA-false-HAL-true,4-MX\_USART2\_UART\_Init-USART2-false-HAL-true,5-MX\_USART1\_UART\_Init-USART1-false-HAL-true**

**RCC.ADCFreqValue=32000000**

**RCC.AHBFreq\_Value=64000000**

**RCC.APB1CLKDivider=RCC\_HCLK\_DIV2**

**RCC.APB1Freq\_Value=32000000**

**RCC.APB1TimFreq\_Value=64000000**

**RCC.APB2Freq\_Value=64000000**

**RCC.APB2TimFreq\_Value=64000000**

**RCC.FCLKCortexFreq\_Value=64000000**

**RCC.FamilyName=M**

**RCC.HCLKFreq\_Value=64000000**

**RCC.IPParameters=ADCFreqValue,AHBFreq\_Value,APB1CLKDivider,APB1Freq\_Value,APB1TimFreq\_Value,APB2Freq\_Value,APB2TimFreq\_Value,FCLKCortexFreq\_Value,FamilyName,HCLKFreq\_Value,MCOFreq\_Value,PLLCLKFreq\_Value,PLLMCOFreq\_Value,PLLMUL,RTCClockSelection,RTCFreq\_Value,SYSCLKFreq\_VALUE,SYSCLKSource,TimSysFreq\_Value,USBFreq\_Value,VCOOutput2Freq\_Value**

**RCC.MCOFreq\_Value=64000000**

**RCC.PLLCLKFreq\_Value=64000000**

**RCC.PLLMCOFreq\_Value=32000000**

**RCC.PLLMUL=RCC\_PLL\_MUL16**

**RCC.RTCClockSelection=RCC\_RTCCLKSOURCE\_LSE**

**RCC.RTCFreq\_Value=32768**

**RCC.SYSCLKFreq\_VALUE=64000000**

**RCC.SYSCLKSource=RCC\_SYSCLKSOURCE\_PLLCLK**

**RCC.TimSysFreq\_Value=64000000**

**RCC.USBFreq\_Value=64000000**

**RCC.VCOOutput2Freq\_Value=4000000**

**SH.GPXTI13.0=GPIO\_EXTI13**

**SH.GPXTI13.ConfNb=1**

**USART1.BaudRate=9600**

**USART1.IPParameters=VirtualMode,BaudRate**

**USART1.VirtualMode=VM\_ASYNC**

**USART2.BaudRate=9600**

**USART2.IPParameters=VirtualMode,BaudRate**

**USART2.VirtualMode=VM\_ASYNC**

**VP\_SYS\_VS\_Systick.Mode=SysTick**

**VP\_SYS\_VS\_Systick.Signal=SYS\_VS\_Systick**

**board=NUCLEO-F103RB**

**boardIOC=true**